

This listing of claims will replace all prior versions, and listings, of claims in the application.

**Listing of Claims:**

1. – 20. (Canceled)

21. (Currently Amended) A computer readable medium bearing instructions comprising:

instructions for providing an application programming interface (“API”) for an application that requests a database operation, said API comprising:

instructions for an initialization method that is invoked by said application to request said database operation, wherein said initialization method initiates a return communication to said application without waiting for any external event, said return communication allowing the application to continue executing without waiting for results from said database operation;

instructions for signaling an application when a database operation is complete; and

instructions for a finalization method that can be invoked by said application, in response to said signaling, to obtain any results of said database operation, wherein upon invocation of said finalization method, said finalization method:

prepares database results for said application; and  
returns said database results to said application.

22. (Canceled)

23. (Currently Amended) A computer readable medium according to ~~claim 22~~ claim 21, wherein said return communication comprises an event object that can be signaled when said database operation completes.

24. (Currently Amended) A computer readable medium according to ~~claim 22~~ claim 21, wherein said return communication comprises an object with a Boolean flag that can be signaled when said database operation completes.

25. (Currently Amended) A computer readable medium according to ~~claim 22~~ claim 21, wherein said initialization method maintains a callback function for notifying a thread when said database operation completes.

26. - 27. (Canceled)

28. (Previously presented) A computer readable medium according to claim 21, wherein said API further comprises instructions for an open connection method that allows a client application to open a database connection asynchronously.

29. (Previously presented) A computer readable medium according to claim 21, wherein said API further comprises instructions for an execute Structured Query Language (“SQL”) statement method that allows a client application to execute a SQL statement asynchronously.

30. (Previously presented) A computer equipped with software implementing an asynchronous database Application Programming Interface (“API”) allowing a client application to open a database connection asynchronously, said API comprising:

instructions for an initialization method, wherein said initialization method is invocable by a client thread to request a database operation, wherein upon invocation said initialization method:

initiates a return communication with the client thread without waiting for any external event;

validates input parameters for a database operation;

sets up a database operation by generating database instructions based on data passed to the initialization method, said database operation comprising at least one instruction to execute a Structured Query Language (“SQL”) statement; and

sends a request for said database operation to a database server;

instructions for signaling said client thread when said database operation is complete;

instructions for a finalization method that can be invoked by said client thread, in response to said signaling, to obtain any results of said database operation, wherein upon invocation said finalization method:

prepares database results for said client thread; and  
returns said database results to said client thread; and  
wherein said instruction for an initialization method and said instructions for a finalization method are stored on a computer readable medium accessible by said computer.

31. (Previously presented) The computer of claim 30, wherein said return communication comprises an event object that can be signaled when said database operation completes

32. (Previously presented) The computer of claim 30, wherein said return communication comprises an object with a Boolean flag that can be signaled when a database operation completes.

33. (Previously presented) The computer of claim 30, further comprising instructions for a callback function for notifying said client thread when said database operation completes.

34. (Previously presented) A computer-enabled method for allowing a client application to asynchronously access database services, said method comprising:

receiving, by Application Programming Interface (API) software executing on a computer, a client thread invocation of an initialization method, wherein said client thread invocation comprises input parameters for a database operation;

initiating, by said initialization method without waiting for any external event, a return communication to said client thread;

validating, by said initialization method, said input parameters;

setting up said database operation by generating database instructions based on said input parameters, said database operation comprising at least one instruction to execute a Structured Query Language (“SQL”) statement;

sending a request for said database operation to a database server;

signaling said client thread when said database operation is complete;

receiving, by said API, an invocation of a finalization method from said client thread in response to said signaling;

preparing, by said finalization method, database results for said client thread; and

returning, by said finalization method, said database results to said client thread.

35. (Previously presented) The method of claim 34, wherein said return communication comprises an event object that can be signaled when said database operation completes

36. (Previously presented) The method of claim 34, wherein said return communication comprises an object with a Boolean flag that can be signaled when a database operation completes.

37. (Previously presented) The method of claim 34, further comprising notifying said client thread when said database operation completes by sending a callback function to said client thread.